REMARKS

The Examiner's Action mailed on July 2, 2007, has been received and its contents carefully considered. Reconsideration of the final rejections presented therein is requested for at least the following reasons, and an RCE is submitted herewith under 37 CFR §1.114. Additionally attached to this Amendment is a Petition for a One-month Extension of Time, extending the period for response to November 2, 2007.

In this Amendment, Applicants have amended claims 1, 7 and 9. Claims 1 and 9 are the independent claims, and claims 1-14 remain pending in the application. For at least the following reasons, it is submitted that this application is in condition for allowance.

Claims 1-8 were rejected under 35 U.S.C. §103(a) as being obvious over *Duvvury* (U.S. 2005/0213560 A1) in view of *Anderson et al.* (U.S. 2003/0217123 A1). This rejection is respectfully traversed.

As amended, claim 1 recites that "said component parts of said Ethernet switch are formed on said single chip".

Amended claim 1 also recites "an address resolution control logic including a source address learning engine for performing a packet source address learning process under the daisy chain test mode to deliver a test packet through the plurality of ports progressively from a start transmission port to a stop receiving port to test the chip". See page 2, lines 19-23 of the specification for support:

> An address resolution control logic in the chip operates a source address (SA) learning engine equipped with the function of the daisy chain test for an inputted or internally generated test packet to be eventually delivered from the STP to the SRP so as to test the chip.

In *Duvvury*, a group of network devices, such as Ethernet switches, are logically configured as a single cluster, with one commander device and one or more member devices. The devices of the cluster are arranged in a star topology, a daisy-chain topology, or daisy-chained off a star topology. See the Abstract and ¶ [0075]:

> **Abstract**
>
> A group of network devices, such as Ethernet switches, are logically configured as a single cluster, with one commander device and one or more member devices. Each network device in the cluster contains an embedded HTML server that facilitates configuration and management of the network device via a management station running a Web browser. Each device in the cluster is identified by a unique Universal Resource Locator ("URL"). However, only the cluster commander is required to have a public IP address. The cluster commander automatically assigns private IP addresses to the other devices in the cluster. Network devices in the cluster constantly monitor network traffic on all their ports to detect conflicts between the automatically assigned IP addresses and the IP addresses of network devices outside of the cluster. When a conflict is detected, the cluster commander assigns a different private IP address to the cluster network device that caused the conflict. The process of detecting and correcting IP address conflicts continues continuously to enable the cluster network devices to react automatically to network configuration changes.

> [0075] The method of creating a cluster of Ethernet switches depends on each particular network configuration. If the switches are arranged in a star topology, as in FIG. 8, with the commander switch at the center, all of the member switches may be added to the cluster at once. On the other hand, if the switches are connected in a daisy-chain topology, as in FIG. 9, the candidate switch that is connected to the commander switch is added first, and then each subsequent switch in the chain is added as it is discovered by CDP. If switches are daisy-chained off a star topology, as in the exemplary hybrid configuration shown in FIG. 10, all the switches that are

> directly connected to the commander switch may be added first, and then the daisy-chained switches may be added one at a time.

Switching alleviates congestion in Ethernet, Token Ring, and Fiber Distributed Data Interface (FDDI) and other similar LANs by reducing traffic and increasing bandwidth. See ¶ [0013]:

> [0013] Switching is a technology that alleviates congestion in Ethernet, Token Ring, and Fiber Distributed Data Interface (FDDI) and other similar LANs by reducing traffic and increasing bandwidth. LAN switches are designed to work with existing media infrastructures so that they can be installed with minimal disruption of existing networks.

The Ethernet switch includes a Layer 1 Physical Interface ("PHY"), a Layer 2 Media Access Control Interface ("MAC"), a frame buffer memory, a source address table memory, a discovery protocol logic, a learning logic, a forwarding logic, a packet redirection logic, and a configuration and management interface. During operation, the learning logic will look at the source address ("SA") within a received Ethernet frame and populate the Source Address Table ("SAT") memory with three columns: MAC address, port number, and age. See ¶¶ [0019] and [0020]:

> [0019] Referring now to FIG. 2-B, a block diagram of an -Ethernet switch according to one aspect of the present invention is shown. As shown in FIG. 2-B, Ethernet switch **200** includes a Layer 1 Physical Interface ("PHY") **202**, **204**, and a Layer 2 Media Access Control Interface ("MAC") **206**, **208**, for each port on the Ethernet switch **200**. A network interface card ("NIC") consists of a MAC and a PHY. An Ethernet switch also contains a MAC and PHY on every port. Thus, an Ethernet switch may appear to a network as multiple NICs coupled together. Each switch PHY **202**, **204**, receives the incoming data bit stream and passes it to its

corresponding MAC **206**, **208**, which reassembles the original Ethernet frames.

[0020] Ethernet switch **200** also includes a frame buffer memory **210**, **212**, for each port, a source address table memory **220**, discovery protocol logic **230**, learning logic **240**, forwarding logic **250**, packet redirection logic **260**, and a configuration and management interface **270**. During operation, the learning logic **240** will look at the source address ("SA") within a received Ethernet frame and populate the Source Address Table ("SAT") memory **220** with three columns: MAC address **280**, port number **282**, and age **284**. The MAC address is the same as the source address that a sender has embedded into the frame. The age item will be a date stamp to indicate when the last frame was received from a particular MAC SA. In the example shown in FIG. 2-B, the port number may be 1 or 2. The SAT is also known as the Switch Forwarding Table ("SFT").

Neither in the abstract, nor in any of ¶¶ [0013], [0019], [0020] or [0075], nor in any other part of *Duvvury* is there any teaching or suggestion that "said component parts of said Ethernet switch are formed on said single chip" as recited in claim 1. Also, in *Duvvury*, the Ethernet switches of the cluster are connected in a daisy-chain topology rather than switched to a daisy chain test mode. An Ethernet switch (for example the commander switch) switches the other Ethernet switches (for example the member switches) to a daisy chain, rather than "a switch for switching the Ethernet switch to a daisy chain test mode" as recited in claim 1.

The Office Action admits that *Duvvury* fails to disclose the address resolution logic of the invention, and alleges that this is taught by ¶ [0573] of *Anderson et al.*

In *Anderson et al.*, the modules are daisy chained together. See FIG. 10 and ¶ [0449]:

> [0449] Normally NET-911 Control Modules are connected to one of the KEY-VIEW PC's serial port (FIG. 12). These modules are daisy chained together (FIG. 10) and connected to each Host PC accessible by the KEY-VIEW PC to permit serial access to and power control of the Host PCs by a remote user. The serial access feature of the modules is normally used to facilitate file transfers between a Host PC 201-204 and KEY-VIEW PC 200. Any other device for which serial access and/or power control is required remotely, such as routers or printers, may be connected to modules on the daisy chain. So, in this regard the KEY-VIEW PC becomes an all inclusive platform for remotely managing network devices.

Further, in *Anderson et al.*, when modules are first installed on the daisy-chain, it is important to test that the module functions properly by accessing the module, toggling power to the device connected to the module, and testing that whatever is connected to the module's serial port can be successfully accessed serially, see ¶ [0573]:

> [0573] When modules are first installed on the daisy-chain, it is important to test that the module functions properly by accessing the module, toggling power to the device connected to the module, and testing that whatever is connected to the module's SERIAL port can be successfully accessed serially.

That is, the daisy chain is a connected configuration rather than a test mode, and the module functions are tested by accessing whatever is connected to the module's serial port rather than delivering a test packet from the start transmission port (STP) to the stop receiving port (SRP).

Thus, *Anderson et al.* does not teach or suggest "an address resolution control logic including a source address learning engine for performing a packet source address learning process under the daisy chain test mode to deliver a test

packet through the plurality of ports progressively from a start transmission port to a stop receiving port to test the chip" as recited in claim 1.

Further, *Anderson et al.*'s disclosed system and method are for accessing and operating personal computers remotely, rather than a Ethernet switch, *Anderson et al.* never mentions Ethernet switches, and the term "Ethernet switch" never appears in *Anderson et al.*

Consequently, as neither *Duvvury* nor *Anderson et al.*, whether taken separately or in combination, teaches or suggests "an address resolution control logic including a source address learning engine for performing a packet source address learning process under the daisy chain test mode to deliver a test packet through the plurality of ports progressively from a start transmission port to a stop receiving port to test the chip" as recited in claim 1, claims 1-8 are allowable.

Claims 9-14 were rejected under 35 U.S.C. §103(a) as being obvious over *Chase et al.* (U.S. 2004/0202157 A1) in view of *Duvvury* (U.S. 2005/0213560 A1). This rejection is respectfully traversed.

Claim 9 as amended recites "proceeding a packet source address learning process for delivering the test packet from the start transmission port to the stop receiving port progressively, wherein the step of proceeding employs a source address learning engine with a daisy chain testing function; and determining a test result by verifying a last received packet at the stop receiving port". For support see the specification, page 3, lines 3-8:

> ... a packet SA learning process to guide the test packet from the STP to the SRP under the operation of the source address learning engine with the function of daisy chain test, and determining the test result by verifying the last received packet at the SRP.

In *Chase et al.*, an Ethernet protocol Metropolitan Area Network (MAN) comprises a plurality of Multi-Service Platforms (MSPs), and each MSP takes the form of an Ethernet switch or the like. A fiber ring or SONET ring infrastructure connects the platforms in daisy-chain fashion allowing each MSP to statistically multiplex information onto, and to statistically de-multiplexing information off the ring infrastructure, see ¶ [0017]:

> [0017] FIG. 1 depicts an Ethernet Protocol Metropolitan Area Network (MAN) **10** comprised of a plurality of Multi-Service Platforms (MSPs) **12_1-12_n** where n is an integer, each MSP taking the form of an Ethernet switch or the like. In the illustrated embodiment n=4 although the network **10** could include a smaller or larger number of MSPs. A fiber ring or SONET ring infrastructure **14** connects the platforms **12_1-12_4** in daisy-chain fashion allowing each MSP to statistically multiplex information onto, and to statistically de-multiplexing information off the ring infrastructure **14**.

Chase et al. only mentions a network including a plurality of Ethernet switches connected in daisy-chain fashion, and *Chase et al.* does not relate to a technique for testing an Ethernet switch. More specifically, Chase et al. relates to an entire network rather than an Ethernet switch. Furthermore, the information frames of Chase et al. are sent between the different devices, such as between the premises, the MSP, the ATM, and the recipients, rather than the plurality of ports within the same device. See FIG. 5 and ¶ [0028]:

> [0028] In the above-described embodiments, the frames of customer traffic have been assumed to comprise IP packets that terminate on a router (i.e., Provider Edge Routers **18**, **180** and **1800**) and that the VPNs employ MPLS-BGP protocols. However, some customers may require multi-protocol support, or may otherwise require conventional PVCs so that the traffic streams must be mapped into Frame Relay or ATM PVCs as depicted in FIG. 5, which illustrates a portion of a MAN **10000** having a CO MSP$\mathbf{12000_4}$ serving an ATM switch **30** that receives traffic from the MAN. As seen in FIG. 5, each of premises $\mathbf{16000_1}$, $\mathbf{16000_2}$ and $\mathbf{16000_3}$ belonging to customer **1**, customer **2** and customer **3**, respectively 5 may originate information frames for receipt at MSP $\mathbf{12000_2}$ in the MAN **10000**. The MSP $\mathbf{12000_2}$ tags each frame with the corresponding customer descriptor prior to statistically multiplexing the data for transmission on the fiber ring infrastructure **14** to the CO MSP $\mathbf{12000_4}$ for receipt at the ATM switch **30**. The ATM switch 30 then maps the frame to the appropriate PVC in accordance with the customer descriptor in the frame in a manner similar to the mapping described with respect to FIG. 3. Thus, the ATM switch **30** could map the frame to one of Frame Relay recipients' $\mathbf{32_1}$, $\mathbf{32_2}$, or $\mathbf{32_3}$; ATM recipients $\mathbf{32_4}$ or $\mathbf{32_5}$ or IMA (Inverse Multiplexing over ATM) recipient $\mathbf{32_6}$.

That is, *Chase et al.* does not teach or suggest "proceeding a packet source address learning process for delivering the test packet from the start transmission port to the stop receiving port progressively, wherein the step of proceeding employs a source address learning engine with a daisy chain testing function; and determining a test result by verifying a last received packet at the stop receiving port" as recited in claim 9.

The Office Action admits that *Chase et al.* does not disclose an "address learning process for delivering the test packet from the start transmission port to the stop receiving port progressively", and alleges that this is taught in ¶ [0023] of *Duvvury*.

In *Duvvury*, the packets are sent between the network devices in a cluster configuration, such as between the commander and member switches via the

network connection, rather than between the ports of an Ethernet switch, see ¶¶ [0023] and [0086]:

> [0023] According to embodiments of the present invention, discovery protocol logic **230** receives, processes, and sends Cisco Discovery Protocol ("CDP") or other discovery protocol packets to neighboring network devices on the network. Packet redirection logic **260** examines the source and destination addresses of Ethernet packets under control of the configuration and management interface **270** and forwards them to other network devices in a cluster configuration. As known to those skilled in the art, the program code corresponding to discovery protocol logic **230**, learning logic **240**, forwarding logic **250**, packet redirection logic **260**, configuration and management interface **270**, and other necessary functions may all be stored on a computer-readable medium. Depending on each particular application, computer-readable media suitable for this purpose may include, without limitation, floppy diskettes, hard drives, RAM, ROM, EEPROM, nonvolatile RAM, or flash memory.
>
> [0086] Configuration and management data packets are sent between the commander **100** and member switches **102-A-102-N** via the network connection. The commander **100** identifies each member switch **102-A-102-N** by the MAC address of the port on the member switch that is connected to the commander **100**. FIG. 12 illustrates in block diagram form how a packet intended for a member switch is processed by the commander. A command from the management station **104** is received by the Ethernet module **122** of the commander switch **100**. The command is processed at the IP layer **124**, UDP or TCP layer **126**, and Management Application layer **128** of the commander switch **100**. The Management Application layer **128** determines that the command is intended for member switch **102**, and performs redirection by translating the port number in the received command to the appropriate port for member switch **102**. The redirected command flows down through the UDP or TCP layer **126**, the IP layer **124**, and the Ethernet layer **122** of the commander switch **100**, and is passed on via Ethernet to the member switch **102**.

In *Duvvury*, the ports belong to different switches, and a packet destination address is determined based on the command from the management station. In this application, the ports belong to the same switch, and the learning process sets a packet destination address as a next port.

Therefore, *Duvvury* also does not teach or suggest "proceeding a packet source address learning process for delivering the test packet from the start transmission port to the stop receiving port progressively, wherein the step of proceeding employs a source address learning engine with a daisy chain testing function; and determining a test result by verifying a last received packet at the stop receiving port" as recited in claim 9.

Consequently, neither *Chase et al.* nor *Duvvury*, whether taken separately or in combination, teaches or suggests the above feature of claim 9, and therefore claims 9-14 are also allowable.

It is submitted that this application is in condition for allowance. Such action and the passing of this case to issue are requested.

Should the Examiner feel that a conference would help to expedite the prosecution of this application, the Examiner is hereby invited to contact the undersigned counsel to arrange for such an interview.

Should the remittance be accidentally missing or insufficient, the Commissioner is hereby authorized to charge the fee to our Deposit Account No. 18-0002, and advise us accordingly.

Respectfully submitted,

October 26, 2007
Date

Alun L. Palmer – Reg. No. 47,838
RABIN & BERDO, PC – Cust. No. 23995
Facsimile: 202-408-0924
Telephone: 202-371-8976

ALP/pq